US-PAT-NO: 6516350

DOCUMENT-IDENTIFIER: US 6516350 B1

TITLE: Self-regulated resource management of distributed computer resources

---------- KWIC ---------

Abstract Text - ABTX (1):

A distributed system for the management of distributed resources interconnected by a computer network and comprised of a plurality of limited supply resources (such as those associated with multimedia content servers, e.g., bandwidth, CPU, storage, etc.,). The system comprises a plurality of server **resources,** preferably with target multimedia content, real time processing engines, etc., and a plurality of intermediary system **resources** (such as directories and **resource monitors**). The resources in any particular server are classified or partitioned as global or local, where the ratio of global to local is specified and controlled by the system. The system assigns clients across local and global resources and coordinates the placement of replicas of target content across global resources. The placement is dynamic and performed when necessary based on the analysis of **utilization** patterns of target content and replicas by pluralities of clients.

Application Filing Date - AD (1):

19990617

Brief Summary Text - BSTX (3):

The present invention relates generally to techniques for managing distributed multimedia **resources** on wide area networks characterized by large bandwidth and **resource** reservation capabilities, and, more specifically, to a system that provides the ability to map, reserve, **monitor,** and manage a dynamic plurality of computing and storage **resources** via a collection of arbitrators, **policies** and directories.

Brief Summary Text - BSTX (7):

The emerging Internet relies on the bandwidth, which is on the order of several magnitudes larger than current Internet provides. It also alleviates network **resource** management and QoS control by introducing correspondent reservation and **monitoring** mechanisms. However, it is clear, that to date, mechanisms for the collective management of multiple media connections that efficiently leverage the sharing of resources across multiple servers in a wide area network are not found.

Brief Summary Text - BSTX (15):

A new breed of high performance applications such as remote surgery, robotics, tele-instrumentation, automated crisis response, digital libraries of satellite data, distance learning via multimedia supported Web sites, enhanced audio, and video, is emerging. However, to accommodate such high performance applications and their continuous media flows, it is not enough to increase or reserve network capacity. These new applications require end-to-end **resource** reservation and admission control, followed by co-ordination of distributed functions such as: (a) **resource** scheduling (e.g., CPU, disk, etc.) at the end-system(s), (b) packet scheduling and flow control in the network, and (c) **monitoring** of the delivered end-to-end quality of service. It is essential that quality of service is configurable, predictable and maintainable system-wide, including the end-system devices, communications subsystem, and networks. Furthermore, all end-to-end elements of distributed systems architecture must work in unison to achieve the desired application level behavior.

Brief Summary Text - BSTX (19):

Thus, the architectures mentioned above do not provide a coordinated management of overall system resources as a function of request activities for individual content and computing resources. Consequently, quality of service must be degraded in response to growing volume of requests for such services over and above an established limit. As the above-mentioned architectures focus on providing QoS as requested by application, they do not take an advantage of a possible aggregation of resources due to commonality between user requests for a particular service. For example, it would be desirable to determine commonality for the usage history of a particular multimedia content, e.g., bursts of requests within short time intervals, the proximity of origination addresses of requests, etc. In addition, the architectures described above do not allow for dynamic **monitoring** and recording of **resource** consumption for individual services as well as for groups of related services, with the purpose of calculating cost of service for individual clients.

Brief Summary Text - BSTX (24):

It is a further objective of the invention to provide a resource management system that self-regulates, according to **policies,** a plurality of end resources and server-based capabilities.

Brief Summary Text - BSTX (26):

According to the principles of the invention, the service control plane is an intelligent intermediary decision plane for managing and enforcing end-to-end properties in the availability of end resources in a computer networked environment. The service control plane receives stimuli (i.e., application requests) from a plurality of clients desiring to access target multimedia content, for instance, and responds to these with end-to-end mapping recommendations satisfying one or more properties (**policies**) set in the service control plane. When a client accepts a mapping recommendation, the service control plane guarantees its performance as set forth by the **policies** used to generate it.

Brief Summary Text - BSTX (29):

Global resources may be made available as a function of integrated demand from clients. To satisfy the requests of clients requesting access to target content, global server-side resources are allocated. A plurality of intermediary system **resources** (such as directories and **resource monitors**) enable the management of such requests for target content in a process referred to as mapping, brokering and negotiation.

Brief Summary Text - BSTX (30):

According to one aspect of the disclosure, the capacity of global resources is dynamic and performed when deemed necessary based on the analysis of **utilization** patterns of target content and replicas by pluralities of clients. The system dynamically shapes capacity by (a) controlling the number of replicas associated with an object and (b) the placement of these replicas.

Brief Summary Text - BSTX (32):

Still another aspect of the invention relates to management of the end-resources. A skilled artisan will appreciate that today resource management is not feasible in part because interfaces to end systems are closed and proprietary. According to the system and method of the invention, interfaces and mechanisms are provided that facilitate **resource monitoring,** setup, and management.

Brief Summary Text - BSTX (36):

Advantageously, the system and method of the invention provides important enhancements to **resource** management architectures not yet available today: 1) it provides a highly available distributed decision capability that may be used to achieve and enforce properties over the usage, as well as patterns of usage, of end-**resources**; 2) it creates a **resource monitoring** point with respect to the performance of multiple end **resources** and clients and their usage patterns so as to provide parameters on where, when, and how to satisfy a request; and 3) it provides an insertion point that can be used for dynamic insertion of both **resources** and content in a manner transparent to clients.

Detailed Description Text - DETX (6):

The contemporary Internet streaming applications establish individual flow between a client and a server, taking care of the QoS, security, encoding and all other issues on per-flow basis. In order to provide means to intervene into this process, a Service Control Plane (SCP) middleware is provided such as shown in FIG. 4. The SCP (400) is located between the clients (121, . . . ,124) and the server resources (111, . . . ,114), interrupts client requests, finds their rate, density and proximity, determines what content is "hot" and predicts the distribution of the requests for such "hot" content. On the other hand, the SCP (400) **monitors** the availability of the **resources,** maps the requests to the servers with available **resources,** predicts **utilization** of the end-**resources** and if necessary, dynamically re-distributes the content. Because the SCP enables borrowing of under used resources from servers (113) and (114) for increasing the resources of the server (111), for example, the resources at the server 111 grow sufficiently and on-time to satisfy

expeditiously all the requests to the content on this server. The SCP also enables returning resources back to the lending servers when the rate of requests decreases.

Detailed Description Text - DETX (10):

Specifically, the Service Management Layer (600a) provides management of the per-flow connection by receiving requests for objects and mapping those requests to the particular server(s) based on factors such as: aggregated demand statistics, and willingness of the servers to provide the requested services. A more detailed description of this aspect of the invention is provided in commonly-owned, co-pending U.S. patent application Ser. No. 09/335,272, entitled SYSTEM AND METHOD FOR INTEGRATED LOAD DISTRIBUTION AND RESOURCE MANAGEMENT IN AN INTERNET ENVIRONMENT, now U.S. Pat. No. 6,463,454 the whole contents and disclosure of which is incorporated by reference as if fully set forth herein. The Service Management Layer (600a) further provides a mapping of a request to the object, taking in consideration various parameters, such as user preferences embodied by user **policies** (611), client parameters, and also per-flow **policies** embodied as **policies** (612), such as a **policy** for service delivery, including availability, quality, interactivity level, etc.

Detailed Description Text - DETX (11):

The System Management Layer (600b) provides an aggregated demand and capacity management and control by attempting to match predicted demand for web objects to available capacity on web servers by dynamically shaping both demand and capacity based on some criteria established by the system **policies** (652). Those **policies** define set criteria for exploration of the willingness of the servers to provide services, negotiation **policy** between overall demand and placement of replicas, etc. Further details regarding the capacity shaping aspect of the invention may be found in commonly-owned, co-pending U.S. patent application Ser. No. 09/335,273, entitled SYSTEM AND METHOD FOR CAPACITY SHAPING IN AN INTERNET ENVIRONMENT, now U.S. Pat. No. 6,466,980 the whole contents and disclosure of which is incorporated by reference as if fully set forth herein.

Detailed Description Text - DETX (13):

As described in aforementioned herein incorporated co-pending U.S. patent application Ser. No. 09/335,272 now U.S. Pat. No. 6,463,454, the negotiations are based on user profiles, per-flow **policies,** and server and replica directories. User profiles (611) characterize the client resources, such as processing power, memory, available bandwidth, etc., and exemplify user preferences, such as preferable compression, cost limits, interactivity level, e.g., the possibility to control the flow of content in time, previewing set-up, willingness to accept or reject commercial advertisement, etc. The negotiator object (603) uses tentative placements, derived from both a replica directory service (666) (see FIG. 7(a)) and a server directory service (656) (see FIG. 7(b)), and executes a query strategy to query the servers associated with those tentative placements. The resulting placement is provided according to the exploration and negotiation per-flow **policies**. Exploration **policy** defines the query strategy, e.g. how many servers should be queried in parallel. The negotiation **policy** is used to refine multiple tentative

placements and enable choosing of a placement based on criteria including: server resource availability, e.g., a server having a willingness "green" factor, and server capacity, e.g., a server having a "high" available capacity. The combination of these two criteria makes the risk of service rejection minimal over time, and facilitates the overall load **balancing**, than in case of placement to a "red" and low capacity server. Per-flow **policies** allow customizing placement, and could either be loaded during initialization, or on-demand.

Detailed Description Text - DETX (14):

As mentioned, negotiations involve optional per-flow services, including, but not limited to: a security service (621), a billing and pricing service (622), a transcoding service (623), and a streaming service (624). For instance, the security services (621) may be added to provide user authentication and authorization as well as encryption/decryption of data. The pricing service (622) enables media services to be offered at various costs and permit a wide variety of pricing **policies**. The billing service (622) allows the cost of the service to be charged to the service user. The transcoding service (623) may request a real-time compression method, as requested by a profile. The streaming service (624) allows the media servers to conform to the various functions, such as a provision to publish or update media. The security service (621), billing and pricing service (622), transcoding service (623), and streaming service (624) each may plug-in to the service management layer (600a) via custom facades or interfaces 631, 632, 633 and 634, respectively. It should be understood that the list of services is not exhaustive, and others may be added by implementing associated facades.

Detailed Description Text - DETX (15):

Within the System Management Layer (600b) the service objects (604) undergo another negotiation process (643) which is directed to generating and distributing object replicas over available servers. As shown in FIG. 6, this negotiation process takes into account aggregated demand statistics (653), selects the set of "hot", or most popular objects according to system **policies** (652) and server resource profiles (651). For example, FIG. 8 illustrates the distribution of object replicas over the territory of the United States, as determined by the System Management Layer (600b) at a particular instant of time.

Detailed Description Text - DETX (16):

As further shown in FIG. 6, the initial placement of replicas onto servers (671), (672) and (673) is defined by the service provider and (or) content owner, and reflected in the replica directory (666) and server directory (656). The server brokers (605) reflect the willingness of the servers to accept the service, which is monitored and controlled locally by the server itself, and reported as **utilization** state indicated as Red, Yellow or Green states into the server directory (656) (see FIG. 7(b)). For example, if the local server (671) holds the replica which is requested by the client (601), and the condition of the server (671) is "Green," the Service management layer may bind the client (601) and the replica on the server (671), establishing the streaming flow (680) between them. After the binding between the client and a server is

established, the QoS control, interactivity control, etc., is further provided by the point-to-point application itself. However, if the per-flow QoS parameters for an individual request cannot be delivered, then the control returns back to the System Management Layer, which remaps the request to another server.

Detailed Description Text - DETX (23):

The relevant disclosure described in greater detail in found in commonly-owened, co-pending U.S. patent application Ser. No. 09/335,272 now U.S. Pat. No. 6,463,454 describes the method of overall **resource monitoring** for the case when several applications run in parallel. However, it is assumed that service bins are additive. In other words, the overall service resource consumed by several applications that are running in parallel is equivalent to a sum of resource bins required by each application. The overall storage is always a sum of storage bins, unless the applications share the same content, and therefore require a common storage bin.

Detailed Description Text - DETX (24):

A system administrator configures overall resources as local and global. The administrator is responsible for establishing the ratio of local to global resources for each server as well as to establishing **policies** of "green" "yellow" and "red" load limits for those resources. After configuring a partition as global storage, the global resource management takes over the control of this resource. Thus, a global storage bin represents a partition that can only be reserved by the global resource management provided by SCP. Note, that the system administrator or the server itself, depending on a relevant **policy,** may re-claim the global resource in full or partially, by requesting its release from the SCP management. The native reservation management process (1020) as shown in FIG. 10, is responsible for **monitoring resource** consumption, and determining the server willingness to accept new requests. It is understood that, the native process may discriminate between application requests for global and local content. Additionally, a request for placement of a global replica may be declined or accepted, as controlled by a pre-installed **policy** that enables maximization of the cost of global resources that may be billed to a global resource requester. Such a **policy** may also depend on request statistics monitored by the SCP.

Detailed Description Text - DETX (30):

In another implementation, each independent content provider runs a server referred to as a delegate controller behind its firewall. The delegate controller performs demand and capacity monitoring tasks for the content provider and when critical **thresholds** are exceeded, the delegate controller places a replication request to the ISP's controller. The ISP controller then arbitrates between pending replication requests and determines which replication requests get allocated to which global servers. In particular, the same global server may be shared across different delegate controllers.

Detailed Description Text - DETX (35):

In conjunction with the SCP, a final aspect of the invention is directed to

a Resource Management Framework (RMF) which is an extensible, distributed, **policy**-based, object-oriented management framework that supports per-flow media session setup and control between one or more media servers and end users and may augment the SCP. This extensible framework is described in greater detail in commonly-owned, co-pending U.S. patent application Ser. No. 09/335,275 which is currently pending, entitled **POLICY**-BASED MULTIVARIATE APPLICATION-LEVEL QoS NEGOTIATION FOR MULTIMEDIA SERVICES, the whole contents and disclosure of which is incorporated by reference as if fully set forth herein. The RMF framework comprises a mandatory core set of basic functions upon which further features can be added. Such new extended functional compositions for the framework include the addition of user profiles and preferences, security (authentication and authorization), pricing and billing, reservations, **resource monitoring** and others. In addition, the **policies,** upon which management decisions and service selections are based, are >pluggable=and new or updated **policies** may be added, to augment or replace existing **policy** and allow new behavior or refinement of existing behavior.

Detailed Description Text - DETX (36):

The RMF is designed to augment the SCP to facilitate the management of the heterogeneous media streaming systems. The low-level network connections, streaming and media-control remain the domain of the media streaming systems. RMF hooks into those lower levels and provides a higher level management and control of those services. For example, the RMF includes QoS mapping down to media level services based on application level requests, but defers the setup of the network connections to the SCP. Where media streaming may be adaptively rate controlled and managed by the media server, within the bounds permitted by the QoS specification, then the media server can act autonomously. When it is unable to maintain the required QoS level then an exception is raised to the SCP, which may then, according to contact **policies** involved, attempt to expand or migrate the end-resources, and then re-map the clients requests, allowing them to receive services elsewhere.